

PENGEMBANGAN APLIKASI BELAJAR MENULIS ALFABET DAN ANGKA UNTUK PERANGKAT MOBILE BERBASIS ANDROID DAN IOS MENGGUNAKAN FRAMEWORK COCOS2

Kovalevsdeno Al Fajr Amiton, KM. Syarif Haryana, Rohmat Nur Ibhahim*
STMIK Mardira Indonesia, Bandung
Corresponding Email: rohmat_nur@stmik-mi.ac.id*

Abstract

This study is about application development of learn how to write alphabet A-Z both uppercase and lowercase letter, number and object word that length is not longer than 5 letters. The application packed into a game where there are images, animations and audio for background music and sound effect. The purpose is to make player not bored while learning how to write. The images theme is colorful, while the audio theme is cheerful. The application was developed for Android and iOS-based device using Cocos2d-x framework that built in C++ language programming. A lot of variations in both Android and iOS device ultimately from screen resolution could become a problem in application development. The solution that was taken is to use a single image that have biggest resolution, so when it's used into a device which have smaller screen resolution, scaling is used for the image asset. In the development, the player can see hint animation as a guide how to write a letter or number. The writing standard refer to handwritingforkids.com website. The game will be over when the player has finished the writing that fit with the sequence that shown by the hint. The game score will be displayed when the game is over, and the highest score will be saved in the application as a record. The score of the game valued by the player writings. In a level, there are obstacles and checkpoints that placed in certain positions. The checkpoint placed inside of the letter or number line. While the obstacle placed on the outside of the letter or number line. The more the obstacle touched, then the more the player score will decrease.

Keywords: *cocos2d-x framework; learning application; game education*

Abstrak

Penelitian ini berkaitan dengan pembuatan sebuah aplikasi belajar menulis huruf alfabet A-Z baik huruf besar maupun kecil, angka dan kata objek yang panjangnya tidak lebih dari 5 huruf. Aplikasi dikemas dalam sebuah permainan dimana terdapat gambar, animasi dan audio untuk background music dan sound effect. Hal ini ditujukan agar pemain tidak merasa bosan dalam belajar menulis. Gambar-gambar yang digunakan bertema colorful, sementara audio yang digunakan bertema riang. Aplikasi dikembangkan untuk perangkat berbasis Android dan iOS menggunakan framework Cocos2d-x yang dibangun dalam bahasa pemrograman C++. Banyaknya variasi device baik perangkat Android maupun iOS terutama dari resolusi layar dapat menjadi masalah dalam pengembangan aplikasi. Solusi yang diambil adalah menggunakan sebuah asset gambar yang memiliki resolusi paling besar, sehingga pada saat dijalankan pada perangkat yang memiliki resolusi layar

yang lebih kecil, dilakukan penskalaan terhadap asset gambar tersebut. Dalam permainan, pemain dapat melihat animasi petunjuk sebagai panduan bagaimana cara menulis sebuah huruf atau angka. Standar penulisan mengacu pada web handwritingforkids.com. Permainan akan selesai apabila pemain telah menyelesaikan tulisan sesuai dengan urutan yang ditampilkan oleh petunjuk. Skor permainan akan ditampilkan saat permainan berakhir, dan skor tertinggi disimpan dalam aplikasi sebagai record. Skor pada permainan dinilai berdasarkan tulisan pemain. Dalam sebuah level, terdapat obstacle dan checkpoint yang ditempatkan pada titik-titik tertentu. Checkpoint ditempatkan pada garis tulisan dari huruf atau angka. Sementara itu obstacle ditempatkan diluar dari garis tulisan huruf atau angka. Semakin banyak obstacle yang tersentuh, maka skor pemain akan semakin berkurang.

Kata kunci: *cocos2d-x framework; aplikasi pembelajaran; permainan pendidikan*

PENDAHULUAN

Belajar menulis adalah hal yang penting bagi pendidikan seorang anak sejak dini, karena menulis adalah ilmu seumur hidup, setiap jenjang pendidikan dan pekerjaan tidak akan lepas dari kegiatan menulis. Terdapat dua hal yang paling dasar dalam menulis secara umum, yaitu menulis alfabet yang terdiri dari huruf A-Z dan angka yang terdiri dari susunan angka 0-9. Setelah menguasai penulisan huruf dan angka tujuan berikutnya adalah untuk menulis kata, dari kata menjadi kalimat, dari kalimat menjadi paragraf, dan seterusnya. Dalam menulis sebuah alfabet atau angka perlu diketahui bagaimana tata cara menulisnya agar penulisan dapat dilakukan lebih efektif. Dengan belajar menulis seorang anak tidak hanya dapat membuat tulisan yang baik dan benar, tetapi juga melatih dalam hal belajar membaca.

Seiring berjalannya waktu, teknologi dapat membantu manusia dalam berbagai hal, belajar menulis adalah salah satunya. Salah satu media yang dapat digunakan untuk belajar menulis adalah perangkat *mobile* yang

menggunakan teknologi layar sentuh. Dari sekian banyak perangkat *mobile* yang ada, *mobile device* berbasis Android dan iOS adalah salah satunya. Dengan menggunakan media perangkat *mobile*, sebuah aplikasi dapat dikembangkan sebagai salah satu sarana belajar menulis. Cukup banyak aplikasi belajar menulis yang ada di Google Play dan Apple Store. Untuk mengembangkan sebuah aplikasi dalam platform perangkat *mobile* yang berbeda cukup sulit dilakukan, selain banyaknya tingkatan sistem operasi dan variasi resolusi layar *device*, masing-masing *platform* memiliki bahasa *native* yang berbeda pula, misalnya perangkat berbasis Android menggunakan Java sebagai bahasa *native*-nya dan perangkat berbasis iOS menggunakan Objective-C sebagai bahasa *native*-nya. Faktor yang menjadi pertimbangan adalah dari segi waktu dan kemampuan bahasa *native* masing-masing *platform*. Untuk mengatasi hal tersebut, solusi yang dapat dilakukan adalah dengan cara menggunakan sebuah *framework* yang memiliki kemampuan untuk melakukan *cross-platform*, salah satu *framework*

cross-platform yang cukup terkenal adalah Cocos2d-x.

Berdasarkan uraian diatas maka perlu untuk dilakukannya pengolahan aplikasi belajar menulis sebagai hal yang menarik dan untuk menjamin aplikasi yang dibuat dapat berjalan di semua tingkatan sistem operasi Android dan iOS dengan resolusi layar yang berbeda. Hal ini dilakukan untuk mengembangkan aplikasi belajar menulis untuk perangkat Android dan iOS menggunakan *framework* Cocos2d-x serta mengembangkan aplikasi yang dapat berjalan dengan berjalan di berbagai macam resolusi layar. Sehingga diharapkan dapat bermanfaat untuk pengguna untuk dijadikan sebagai sarana untuk belajar menulis yang menyenangkan, dan dapat dijadikan bahan penyusunan penelitian yang serupa dan lebih mendalam untuk menyempurnakan hasil penelitian dan komunitas *mobile app developer* dapat dijadikan referensi untuk mengembangkan aplikasi *cross-platform*.

LANDASAN TEORI

Rekayasa Perangkat Lunak

Menurut Pressman (2010:14) rekayasa perangkat lunak adalah pembuatan dan penggunaan prinsip-prinsip keahlian teknik untuk mendapatkan perangkat lunak yang ekonomis yang handal dan bekerja secara efisien pada mesin yang sesungguhnya. Rekayasa perangkat lunak merupakan teknologi yang bertingkat atau berlapis, dimana lapisan teknologi tersebut dijelaskan sebagai berikut:

1. Fokus pada kualitas (*A quality focus*); Pendekatan teknik apapun (termasuk rekayasa

perangkat lunak) harus bersandar pada komitmen organisasi terhadap suatu mutu. Fondasi yang mendukung rekayasa perangkat lunak adalah fokus pada kualitas.

2. Proses (*Process*); Dasar untuk rekayasa perangkat lunak adalah lapisan proses. Proses pada rekayasa perangkat lunak adalah perekat yang memegang teknologi lapisan (*layer*) bersama – sama dan memungkinkan pengembangan perangkat lunak yang rasional dan tepat waktu. Key process areas membentuk dasar kontrol manajemen proyek perangkat lunak dan menetapkan konteks metode-metode teknis mana yang diterapkan, produk kerja (*model*, *dokumen*, *data*, *laporan*, *form*, dan lain-lain) yang diproduksi, *milestone* yang ditetapkan, kualitas yang terjamin dan perubahan yang dikelola dengan baik.
3. Metode (*Method*); Metode rekayasa perangkat lunak menyediakan teknis “bagaimana” untuk membangun perangkat lunak. Metode mencakup tugas yang mencakup analisis kebutuhan (*requirement analysis*), perancangan (*design*), program konstruksi (*program construction*), pengujian (*testing*), dan pemeliharaan (*maintenance*).
4. Alat Bantu (*Tools*); Alat bantu otomatis atau semi-otomatis menyediakan dukungan untuk proses dan metode.

Aplikasi Mobile

Dikutip dari (http://en.wikipedia.org/wiki/Mobile_app), sebuah aplikasi *mobile* adalah aplikasi perangkat lunak yang dirancang untuk berjalan pada *smartphone*,

komputer tablet dan perangkat *mobile* lainnya. Aplikasi *mobile* tersedia melalui *platform* distribusi aplikasi, yang biasanya dioperasikan oleh pemilik sistem operasi *mobile*, seperti Apple App Store, Google Play, Windows Phone Store dan BlackBerry App World. Beberapa aplikasi tersedia secara gratis, sementara yang lain ada yang berbayar. Aplikasi *mobile* bertujuan untuk mempermudah kegiatan sehari-hari, didalamnya termasuk seperti *email*, kalender, kontak, bursa saham hingga informasi cuaca. Namun, permintaan publik dan kemajuan alat pengembangan membuat ekspansi yang cepat kedalam kategori lain, seperti *game mobile*, otomatisasi pabrik, GPS dan layanan berbasis lokasi, perbankan, *order-tracking*, dan pembelian tiket.

Aplikasi

Aplikasi menurut Jogiyanto (1999:12) adalah penggunaan dalam suatu komputer, instruksi (*instruction*) atau pernyataan (*statement*) yang disusun sedemikian rupa sehingga komputer dapat memproses *input* menjadi *output*. Program merupakan kumpulan *instruction set* yang akan dijalankan oleh pemroses, yaitu berupa *software*. Bagaimana sebuah sistem komputer berpikir diatur oleh program ini. Program inilah yang mengendalikan semua aktifitas yang ada pada pemroses. Program berisi konstruksi logika yang dibuat oleh manusia, dan sudah diterjemahkan ke dalam bahasa mesin sesuai dengan format yang ada pada *instruction set*.

Video Game

Video game adalah sebuah *electronic game* yang melibatkan interaksi manusia dengan sebuah *device* yang dihubungkan dengan *user interface* (http://en.wikipedia.org/wiki/Video_game). Menurut (Nilwan, 1996:3) *game* diartikan sebagai suatu aktivitas terstruktur yang biasanya dilakukan

untuk kesenangan dan terkadang digunakan sebagai alat pembelajaran. Komponen utama dari *game* adalah tujuan, aturan, tantangan dan interaksi. *Game* pada umumnya mengikutsertakan stimulasi mental atau fisik, bahkan keduanya.

Perangkat Mobile

Perangkat *mobile* adalah sebuah perangkat komputasi yang biasanya memiliki tampilan layar dengan input sentuhan dan / atau keyboard miniatur dan berat kurang dari 2 pon (0,91 kg). Apple, Samsung, Nokia, Sony, HTC, LG, Blackberry dan Motorola adalah beberapa contoh dari banyak produsen yang memproduksi jenis perangkat *mobile*. Sebuah perangkat *mobile* memiliki sistem operasi (OS), dan dapat menjalankan berbagai jenis perangkat lunak aplikasi. Kebanyakan perangkat *mobile* juga dapat dilengkapi dengan kemampuan Wi-Fi, *bluetooth* dan GPS yang dapat memungkinkan melakukan koneksi ke internet dan perangkat *bluetooth* yang mampu terhubung dengan perangkat keras lainnya. Sebuah kamera atau fitur *mediaplayer* untuk *file video* atau musik juga dapat biasanya ditemukan pada perangkat ini yang juga dilengkapi dengan sumber daya baterai yang stabil seperti baterai *lithium*. (http://en.wikipedia.org/wiki/Mobile_app).

Android

Android adalah sistem operasi berbasis Linux yang didesain khusus untuk perangkat *mobile* berbasis *touch screen* seperti *smartphone* dan komputer tablet. Android dikembangkan oleh Android, Inc. sebelum kemudian diakuisisi oleh Google pada tahun 2005. Android dikenalkan pada tahun 2007. Kemudian untuk mengembangkan Android, dibentuklah Open Handset Alliance, konsorsium dari banyak perusahaan *hardware*, *software* dan telekomunikasi. Android bersifat *open source* dan

Google merilis kode-kode Android dibawah Apache License. Hal ini menjadikan sistem operasi Android dapat bebas dimodifikasi dan didistribusikan oleh para pengembang. Selain itu Android memiliki pasar untuk penjualan aplikasi Android yang bernama Google Play. ([http://en.wikipedia.org/wiki/Android_\(operating_system\)](http://en.wikipedia.org/wiki/Android_(operating_system)))).

iOS

iOS (sebelumnya iPhone OS) adalah sistem operasi perangkat bergerak yang dikembangkan dan didistribusikan oleh Apple Inc. Sistem operasi ini pertama diluncurkan tahun 2007 untuk iPhone dan iPod Touch, dan telah dikembangkan untuk mendukung perangkat Apple lainnya seperti iPad dan Apple TV (<http://id.wikipedia.org/wiki/IOS>). *User-interface* dari iOS menggunakan dasar konsep dari *direct manipulation*, menggunakan *multi-touch gestures*. Kontrol *interface*-nya terdiri dari *sliders*, *switches* dan *buttons*. Interaksi dengan iOS dapat dilakukan dengan *swipe*, *tap*, *pinch* dan *reversepinch*. Ada pula *accelerometer* yang digunakan untuk mendeteksi arah pergerakan dari *device*. Berikut ini adalah versi dari sistem operasi iOS.

Menulis

Menulis adalah representasi bahasa menggunakan media tekstual melalui penggunaan seperangkat tanda atau simbol yang dikenal sebagai sistem penulisan (Daniels, 1996:3). Hal ini berbeda dari ilustrasi, seperti menggambar dan melukis, dan juga pelestarian bahasa non-simbolis melalui media non-tekstual, seperti *audiotape* magnetik. Sekitar milenium ke-4 sebelum masehi, kompleksitas perdagangan dan administrasi di Mesopotamia menjadi sesuatu yang sulit bagi ingatan manusia, dan menulis menjadi metode yang lebih diandalkan

pencatatan dan penyajian transaksi dalam bentuk permanen (Robinson, 2003 : 36).

Alfabet adalah suatu kumpulan standar huruf (simbol tertulis dasar atau grafem) yang digunakan untuk menulis satu atau lebih bahasa yang didasarkan pada prinsip umum bahwa huruf mewakili fonem (suara signifikan dasar) dari bahasa lisan. Ada puluhan huruf yang digunakan saat ini, yang paling umum adalah alfabet Latin (Haarmann, 2004 : 96) yang berasal dari bahasa Yunani. Sedangkan angka adalah objek matematika yang digunakan untuk menghitung, melabeli, dan mengukur. Dalam matematika, definisi angka telah diperpanjang selama bertahun-tahun untuk memasukkan angka seperti nol, angka negatif, angka rasional, angka irasional, dan angka kompleks. Angka adalah suatu tanda atau lambang yang digunakan untuk melambangkan bilangan.

Cocos2d-x

Cocos2d-x adalah *cross-platform game engine* 2D yang paling populer di dunia. Cocos2d-x dilengkapi fitur-fitur yang *powerful*, juga mudah untuk dipelajari dan digunakan. Dan yang paling utama, cocos2d-x gratis dan bersifat *open source*. Cocos2d-x adalah sebuah *game engine* yang dikembangkan dari Cocos2d-iPhone, dimana Cocos2d-x terdiri dari banyak platform yang didukung, dengan banyak pilihan bahasa pemrograman yang mengandung struktur API yang sama. Cocos2d-x pertama kali dikembangkan pada Juli 2010. Cocos2d-x digunakan oleh para individu dan penggemar juga perusahaan besar seperti Zynga, Wooga, Gamevil, Glu, GREE, Konami, TinyCo, HandyGames, IGG, Disney Mobile dan lain-lain. Insinyur dari Chukong, Google, Microsoft, ARM, Intel dan Blackberry aktif terlibat di komunitas Cocos2d-x. (<http://www.cocos2d-x.org/wiki>).

Cocos2d-x dirilis dalam MIT License. Dengan menggunakan *framework* Cocos2d-x, *developer* dapat dengan mudah melakukan *porting game*-nya untuk perangkat berbasis iOS, Android, Windows Phone 8, BlackBerry, Marmalade, WindowsXP / Windows7, Linux, Mac OS X, Windows 8 Metro dan Native Client (NaCl).

OpenGL

Dikutip dari (<http://en.wikipedia.org/wiki/OpenGL>) dan (<http://www.opengl.org/about/>), OpenGL atau *Open Graphic Library* sebuah *multi-platform Application Programming Interface* (API) untuk merender grafik 2D dan 3D. API *OpenGL* biasa digunakan untuk berinteraksi dengan *Graphics Processing Unit* (GPU) untuk mencapai proses render yang berorientasi pada *hardware-accelerated*. OpenGL dikembangkan oleh Silicon Graphics Inc. (SGI) sejak tahun 1991 dan dirilis pertama kali pada tahun 1992. OpenGL umum digunakan pada CAD (*Computer-aided Design*), *virtual reality*, *scientific visualization*, *information visualization*, *flight simulation*, dan *video game*. OpenGL mendorong inovasi dan mempercepat pengembangan aplikasi dengan memasukkan satu set *rendering*, *texture mapping*, *special effects*, dan fungsi visualisasi lainnya. Berikut ini adalah *overview* secara umum alur dari visualisasi OpenGL

METODOLOGI PENELITIAN

Metode penelitian adalah suatu cara atau prosedur yang dipergunakan untuk melakukan penelitian sehingga mampu menjawab rumusan masalah dan tujuan penelitian. Menurut Nasir (1988:51), metode penelitian merupakan cara utama yang digunakan peneliti untuk mencapai tujuan dan menentukan jawaban atas masalah yang diajukan. Metode penelitian yang dilakukan dalam

skripsi ini adalah metode *Research & Development*.

Langkah-langkah *Research & Development* menurut Borg dan Gell (1989:783-795) adalah : 1) *Research and information collecting*; 2) *Planning*; 3) *Develop preliminary form of product*; 4) *Preliminary field testing*; 5) *Main product revision*, 6) *Main field testing*, 7) *Operational product revision*; 8) *Operational field testing*; 9) *Final product revision*; 10) *Dissemination and implementation*.

PEMBAHASAN

Analisis Sistem dan User

Analisis dilakukan untuk menjelaskan kebutuhan sistem dan apa yang harus sistem lakukan untuk memenuhi kebutuhan *user*. Tujuan dari analisis adalah untuk memahami domain masalah dan tanggung jawab sistem dengan mempelajari bagaimana *user* akan menggunakan sistem. Hal ini dapat dilakukan dengan cara membuat beberapa pemodelan dari sistem yang akan dibuat. (Bahrami, 1999:125-126) Analisis *user* atau pengguna bertujuan untuk mendefinisikan siapa yang akan menggunakan aplikasi yang akan dikembangkan, dalam hal ini aplikasi belajar menulis, maka dari itu, diperlukan analisis mendalam mengenai bagaimana karakteristik user terlebih dahulu. User dalam aplikasi belajar menulis yang dikembangkan adalah seseorang yang akan belajar menulis atau pada umumnya anak-anak. Pada setiap anak terdapat kapasitas untuk belajar namun kapasitas di dalam diri anak itu harus di rangsang oleh faktor-faktor yang ada di lingkungannya. Sama seperti tanaman, benih yang bagus bila ditanam pada tanah yang jelek, maka yang tumbuh adalah tanaman yang buruk, bagai tumbuhan mati tak hendak hidup tak mau. Begitu pula anak-anak, bila mereka tumbuh dalam lingkungan yang miskin pendidikan dan miskin

budaya, perkembangan fisik dan mental mereka pun akan terhambat.

Analisis Sistem Permainan

Aplikasi yang akan dibuat menggunakan *touch and drag* sebagai *controller* dalam permainannya. Aplikasi ini bertujuan untuk membentuk huruf, angka atau kata yang ditampilkan dalam permainan dengan cara mengikuti petunjuk yang diberikan. Huruf, angka atau kata dapat dipilih secara bebas. Apabila *user* dapat menyelesaikan suatu huruf, angka atau kata, notifikasi permainan berakhir ditampilkan dan *user* dapat memilih kembali suatu huruf, angka atau kata yang akan dimainkan. Komponen yang paling utama dalam aplikasi yang akan dibuat adalah bagaimana tata cara atau aturan penulisan sebuah huruf alfabet dan angka.



Gambar 1 Standar penulisan huruf alfabet handwritingforkids.com



Gambar 2 Standar penulisan angka handwritingforkids.com

Aturan penulisan diperlukan agar anak yang sedang belajar menulis dapat belajar dan berlatih bagaimana menulis suatu huruf atau angka. Pada awalnya tentu saja harus dibimbing oleh orang yang lebih tua untuk dilakukan pengarahan hingga pada akhirnya anak dapat belajar melakukannya secara mandiri. Sesuai dengan standar penulisan yang telah dijelaskan pada beberapa gambar sebelumnya, hal yang dapat dilakukan adalah membuat beberapa *checkpoint* yang membentuk kumpulan huruf dan angka tersebut.

Aplikasi belajar menulis dibentuk menjadi sebuah *game*, oleh karena itu, ada beberapa hal yang menjadi faktor penunjang agar sebuah *game* menjadi menarik, dua diantaranya adalah audio dan grafik. Dua komponen pendukung dalam *game* akan dijelaskan sebagai berikut; 1) Audio sebuah permainan hampir dapat dipastikan memiliki komponen ini sebagai faktor penunjang permainan. Audio akan dibagi menjadi dua yaitu : *Background music* dan *Sound effect*, 2) Grafik dibuat dalam bentuk permainan 2 dimensi, kebutuhan utamanya mencakup berbagai *image* yang dibungkus dalam sebuah tema permainan.

Analisis Sistem Operasi

Pada sistem operasi Android dan iOS, banyak tingkatan sistem operasi layaknya sistem operasi Windows. Secara umum, perbedaannya terletak pada *user-interface* dan *software* pada sistem operasi sehingga sistem operasi yang paling baru adalah yang lebih baik dibandingkan dengan sistem operasi

yang lama. Dalam segi pengembangan aplikasi, perbedaan yang paling utama adalah tersedianya API (*Application Programming Interface*) pada saat *runtime* atau aplikasi dijalankan. Untuk mengatasi kompatibilitas sistem operasi, cukup dilakukan pengujian secara berkala pada setiap sistem operasi yang ada atau pada sistem operasi versi yang paling lama sehingga diharapkan aplikasi dapat berjalan di semua tingkatan sistem operasi.

Analisis Device

Aplikasi belajar menulis ditujukan untuk dapat berjalan pada sistem operasi Android dan iOS, kendala yang paling mencolok adalah mengenai resolusi layar masing-masing *device*. Terdapat banyak sekali vendor yang mengeluarkan baik *smartphone* ataupun tablet, sebut saja Sony, Samsung, LG, Motorola, HTC, Asus, Acer dan lain-lain yang mengusung sistem operasi Android dimulai dari resolusi yang terkecil yaitu 320x240 pixel hingga terbesar saat ini adalah 2560x1600 dengan *aspect ratio* 3:2 hingga 16:9, demikian pula yang terjadi pada sistem operasi iOS walaupun vendornya hanya satu yaitu Apple. Perbedaan resolusi menjadi salah satu kendala dalam mengembangkan aplikasi yang universal, dalam hal ini peletakan komponen grafik sama pada setiap resolusi agar tampilan aplikasi terlihat rapi, tidak acak-acakan.

Hal berikutnya yang menjadi kendala dalam pengembangan aplikasi khususnya game adalah spesifikasi *device*, dalam hal ini yang ditekankan adalah prosesor, RAM, dan GPU. Dalam aplikasi permainan, umumnya

dibutuhkan kemampuan untuk *render* grafik 2 atau 3 dimensi, hal inilah yang membedakan dengan aplikasi biasa. Kompleksitas pemrograman dan multimedia menjadi faktor utama yang mempengaruhi kinerja *hardware device*, hal ini memunculkan istilah *minimum system requirement*, dimana diperlukan spesifikasi tertentu yang harus dipenuhi oleh *device* agar dapat menjalankan sebuah aplikasi, hal yang terjadi apabila *system requirement* tidak dipenuhi adalah aplikasi akan berjalan lambat atau *lag*, yang paling parah adalah *force close* atau aplikasi tidak dapat dijalankan. Untuk membuat aplikasi yang universal, lagi-lagi diperlukan pengujian secara menyeluruh agar aplikasi dapat berjalan dengan baik terutama dalam manajemen memori dan multimedia.

Analisis Cocos2d-x

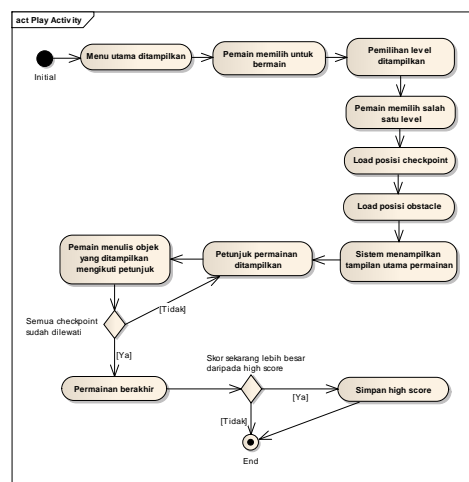
Aplikasi permainan belajar menulis akan dibangun pada *device* berbasis Android dan iOS. Penggunaan *framework* untuk membuat permainan atau yang biasa disebut *game engine* dapat dilakukan dengan pertimbangan kemudahan untuk mengembangkan aplikasi permainan. Kemudahan yang dimaksud adalah ketersediaan *library* untuk penggunaan animasi dan audio sehingga tidak perlu membangun dari nol.

Cocos2d-x adalah sebuah *game engine* yang dikembangkan dalam bahasa C++. Salah satu keunggulan Cocos2d-x adalah kemampuannya untuk dikembangkan dalam berbagai jenis platform dalam satu waktu. Selain itu Cocos2d-x merupakan turunan dari Cocos2d-iPhone, salah satu *game engine open source* yang paling populer dalam

pengembangan aplikasi *game* untuk perangkat berbasis iOS, juga didukung oleh komunitas forum yang besar sehingga memudahkan interaksi antar sesama pengembang aplikasi *game* untuk berbagi dan berkonsultasi.

Evaluasi Sistem

Use case bertujuan untuk menggambarkan interaksi apa saja yang dapat dilakukan oleh aktor dalam permainan belajar menulis yaitu *player*. Activity diagram berikut ini akan menjelaskan *workflow* aplikasi belajar menulis dari segi permainan secara umum



Gambar 3 Activity diagram permainan aplikasi belajar menulis

Perancangan Sistem

Pada perancangan aplikasi belajar menulis, meliputi perancangan *class* menggunakan *class diagram*, perancangan interaksi menggunakan *sequence diagram*, rancangan *graphical user interface* (GUI) hingga penjelasan logika proses berdasarkan analisis pada bab sebelumnya. Tujuannya dari perancangan adalah untuk memberikan

gambaran yang jelas agar mempermudah proses pembuatan aplikasi. (Bahrami, 1999:199-200)

Perancangan Aplikasi

Pada bagian perancangan aplikasi belajar menulis akan dibuat, meliputi perancangan tampilan, penerapan aturan main dan audio dan grafik. Selain sebagai penentuan *win condition*, *sprite* tersebut dapat digunakan sebagai petunjuk penulisan, posisi kelima *sprite* tersebut dibaca untuk kemudian ditampilkan animasi yang bergerak dari posisi *sprite* ke-1 hingga posisi *sprite* ke-5. Animasi akan terus tampil hingga *win condition* telah tercapai dengan patokan *sprite* yang sedang aktif hingga *sprite* terakhir.

Untuk penilaian atau *scoring*, objek *sprite* dibuat sebagai *obstacle* yang akan ditempatkan mengelilingi *font* dari level yang dimainkan oleh pemain. Banyaknya jumlah *obstacle* bervariasi, lebih banyak *obstacle* yang ditempatkan di sekeliling *font* dari level tentu akan lebih baik dengan harapan penilaian penulisan menjadi lebih presisi. Sesuai dengan apa yang dipaparkan mengenai audio, audio dalam aplikasi belajar menulis ini akan dibagi menjadi dua, yaitu *background music* dan *sound effect*. Tema *background music* dan *sound effect* yang dibutuhkan adalah yang bersuasana riang dan menyenangkan.

Dalam aplikasi ini yang ditekankan adalah bagaimana tampilan yang mendukung dalam belajar menulis, untuk itu perancangan grafik diambil sebuah tema colorful sehingga harapannya menarik minat pemain dalam hal ini sasarannya anak-anak

sehingga belajar menulis tidak menjadi hal yang membosankan. Dengan menggunakan asset berdasarkan resolusi yang paling besar, lalu dalam aplikasi dilakukan scaling untuk penyesuaian. Resolusi *default* yang akan digunakan adalah 1920 x 1080 pixel atau yang lebih dikenal sebagai *High Definition* (HD) 1080p.

Perancangan Class

Perancangan *class* menjelaskan atribut dan *method* yang dimiliki *class* yang ada (struktur *class*) pada pembuatan aplikasi belajar menulis dengan menggunakan *class diagram*. Perancangan *class* ini dibuat berdasarkan hasil analisis dan ulasan pada sub-bab sebelumnya, juga disesuaikan dengan *class* yang dimiliki oleh *framework* Cocos2d-x. Pada perancangan kelas dibuat *class - class* beserta atribut dan *method*-nya (struktur data dan perilakunya) yang dapat menjadi elemen-elemen pembangun interaksi yang ada dalam *use case diagram*.

Perancangan Database

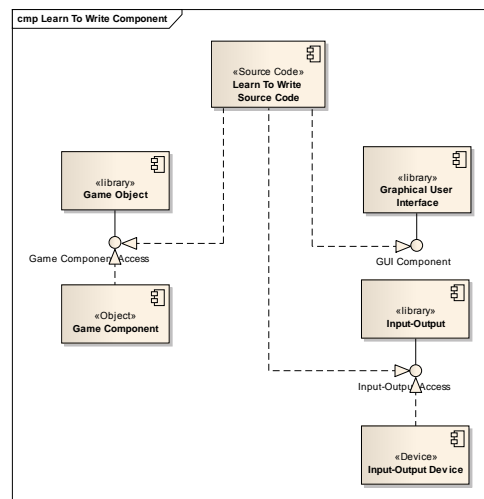
Database yang dirancang pada aplikasi belajar menulis dibuat terintegrasi dengan aplikasi melalui *library* yang disediakan oleh *framework* Cocos2d-x. Pada Cocos2d-x terdapat sebuah *class* yang dapat berfungsi sebagai *database* kecil yaitu *class* CCUserDefault. *Database* yang dibuat dengan memanfaatkan *class* ini akan terintegrasi dengan aplikasi berupa *file* xml. Operasi yang dapat dilakukan sangat sederhana sekali, yaitu perintah *get* dan *set* saja.

Perancangan Sequence Diagram

Sequence diagram digunakan untuk mendeskripsikan bagaimana sebuah proses beroperasi antara satu proses dengan proses yang lain dan bagaimana urutannya. *Sequence diagram* ini dibuat untuk memodelkan perilaku *use case* berdasarkan *object* dan *class* yang telah didefinisikan pada *class*.

Implementasi Dan Pengujian

Untuk pengembangan aplikasi, karena pengembangan aplikasi iOS hanya dapat dilakukan pada perangkat yang bersistem operasi Macintosh menggunakan Xcode, digunakan *software* VirtualBox yang berfungsi untuk memvirtualisasikan sistem operasi Macintosh. Berikut ini adalah *component diagram* yang menggambarkan keterkaitan berbagai *component* yang ada dalam permainan dam-daman.



Gambar 4 Component diagram aplikasi belajar menulis

Dalam mengembangkan aplikasi belajar menulis, digunakan beberapa *library* dari *game engine* Cocos2d-x.



Gambar 5 Tampilan aplikasi

Pengujian

Untuk memeriksa fungsi dan meningkatkan kualitas aplikasi, maka dilakukan pengujian untuk menguji berbagai hal pada aplikasi mulai dari kesesuaian aliran *user interface* hingga *gameplay* dari aplikasi belajar menulis.

Berikut adalah hasil dari pengujian yang telah dilakukan.

1. Pengujian yang dilakukan menggunakan metode *blackbox testing*. *Blackbox testing* adalah sebuah pengujian yang mengesampingkan atau mengabaikan mekanisme internal dari sebuah sistem atau komponen dan berfokus hanya pada *output* yang dihasilkan berdasarkan respon dari *input* dan kondisi eksekusi yang diberikan. (IEEE Standard 610.12-1990 : 1990)
2. *Blackbox testing* yang dilakukan menggunakan pendekatan *system testing*. *System testing* adalah pengujian yang dilakukan pada sistem untuk mengevaluasi kesesuaian sistem berdasarkan spesifikasi *requirement*. (IEEE Standard 610.12-1990 : 1990)
3. Proses pengujian dilakukan terhadap interface aplikasi.
4. Proses pengujian dilakukan terhadap fungsi-fungsi aplikasi.

KESIMPULAN

Menulis adalah ilmu seumur hidup, hal yang mendasar dalam belajar menulis adalah objek tulisan seperti huruf alfabet atau angka dan bagaimana menulis objek tulisan tersebut. Tidak adanya standar yang baku bagaimana cara menulis sebuah huruf atau angka tidak menjadikan belajar menulis menjadi sesuatu yang sulit, karena tujuan utamanya itu sendiri agar seseorang dapat membuat huruf atau angka dengan benar, bagaimana cara menulisnya sebenarnya tidak terlalu menjadi masalah, namun contoh penulisan yang mengacu pada handwritingforkids.com dapat dijadikan sebagai panduan bagi orang yang pertama kali belajar menulis.

Selain tema *art* atau gambar, penambahan efek animasi dan *audio* baik *background music* dan *sound effect* bisa menjadi salah satu elemen yang membuat sebuah permainan menjadi lebih atraktif sehingga belajar menulis menggunakan aplikasi ini diharapkan bisa menjadi hal yang menyenangkan. Untuk membuat aplikasi yang dapat berjalan di sistem operasi iOS dan Android, *framework* Cocos2d-x yang menggunakan bahasa pemrograman C++ cukup bisa diandalkan. Banyaknya variasi resolusi layar tidak menjadi masalah yang berat untuk pengembangan aplikasi, penggunaan *scaling* terhadap setiap *asset* gambar yang digunakan bisa menjadi solusi agar tampilan aplikasi selalu konsisten di pada saat dijalankan di berbagai perangkat, serta yang paling utama adalah pengujian secara berkala dapat menjamin aplikasi berjalan lancar.

REFERENSI

- Bahrami, Ali. 1999. *Object Oriented Systems Development*. Singapore: McGraw-Hill Book Co.
- Borg, Walter R., & Gall, Meredith D. (1989). *Educational Research*. New York: Longman.
- Coulmas, Florian (1989). *The Blackwell Encyclopedia of Writing Systems*. Oxford: Blackwell Publishing.
- Daniels, Peter T; Bright, William. 1996. *The World's Writing Systems*. Oxford: Oxford University Press.
- Haarmann, Harald. 2004. *Geschichte der Schrift*. München: C. H. Beck.
- IEEE. 1990. *IEEE Standard 610.12-1990, IEEE Standard Glossary of Software Engineering Terminology*. New York: The Institute of Electrical and Electronics Engineers.
- Jacobson, Ivar. 1992. *Object Oriented Software Engineering: A Use Case Driven Approach*. Boston: Addison-Wesley Professional.
- Jogiyanto, HM. 2005. *Analisis dan Desain Sistem Informasi Pendekatan Terstruktur*, edisi ketiga. Yogyakarta: Andi Offset.
- Larman, Craig. 2004. *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development*, Third Edition. New Jersey: Prentice Hall.
- Millard, A. R. 1986. *The Infancy of the Alphabet*. London: Taylor & Francis, Ltd.
- Nasir, Mohammad, 1988. *Metode Penelitian*. Jakarta: Ghalia Indonesia.
- Nilwan, Agustinus. 1998. *Pemrograman Animasi dan Game Profesional*. Jakarta: Elex Media Komputindo.
- Parker, Sybil P. 2002. *McGraw-Hill Dictionary of Scientific and Technical Terms*. New York: McGraw-Hill Professional.
- Pender, Tom. 2003. *UML Bible*. New York: John Wiley & Sons, Inc.
- Pressman, Roger S. 2010. *Software Engineering: A Practitioner's Approach*. New York: McGraw-Hill Science/Engineering/Math.
- Robinson, Andrew. 2003. *The Story of Writing*. London: Thames & Hudson.
- Schalkoff, R. 1992. *Pattern Recognition: Statistical, Structural, and Neural Approach*. New York: John Wiley & Sons, Inc.
- Schuermann J. 1996. *Pattern Classification: A Unified View of Statistical and Neural Approaches*. New York: Wiley & Sons.